

HTML

- Basic HTML
- Lists
- Tables
- Input

CSS

- CSS Properties and Values

Course Specific Functions

- SimplePixel
- SimpleImage
- Printing

Standard JavaScript

- Arithmetic Operations
- Comparing Two Numbers
- Combining Comparisons
- Math Functions
- Random Functions

Background Information

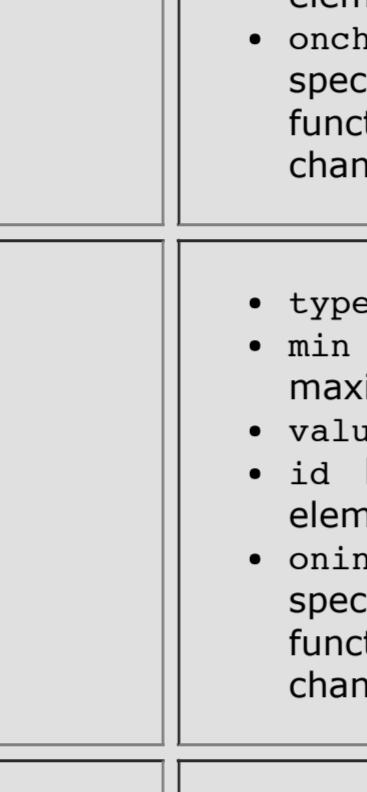
- What is a Pixel?
- Transparency: Alpha Channel
- Image Coordinate System

HTML

Full list of HTML elements

As a reminder, in most browsers you can right-click on a page and select View Source to see the HTML code used to render the page.

Basic HTML

Tag	Description	Example
<code><html></code>	All content of your webpage must go inside <html></html> tags.	
<code><head></code>	Contains information about the webpage.	
<code><title></code>	Title of the webpage (what appears in the window/tab of your browser). The text itself does not appear on webpage.	
<code><body></code>	Everything that appears on the webpage should go between these tags	
<code><p></code>	Defines a paragraph (text with some space on the bottom and top).	<code><p>This is a paragraph.</p></code> This is a paragraph.
<code><h1></code>	Heading tag, bold and bigger text. You can use any number from <code><h1></code> to <code><h6></code> with <code><h1></code> being the largest heading and <code><h6></code> being the smallest.	<code><h1>larger heading</h1></code> <code><h2>smaller heading</h2></code> larger heading smaller heading
<code></code>	Apply bold formatting to text	<code>bold</code> bold
<code></code>	Apply emphasis to text	<code>emphasis</code> emphasis
<code></code>	Inserts an image. <ul style="list-style-type: none">src is the link specifying the image to display (it is a required attribute)width (and height) specifies the size of the image (it is an optional attribute) Unlike most other tags, this start tag does not have a corresponding end tag.	<code></code> 
<code><a></code>	Links to another webpage. <ul style="list-style-type: none">href specifies the URL of the page to link to (it is a required attribute). There must be some text between the start and end tags to be the anchor of the link.	<code>Duke University</code> Duke University
<code><div></code>	Defines a section of the web page.	<code><div><p>This paragraph is inside a div.</p></div></code>

Lists

Tag	Description	Example
<code></code>	List item. List items can go inside unordered list, <code></code> , or ordered list, <code></code> tags.	<code>HTML</code>
<code></code>	Unordered list, each item has a bullet point.	<code></code> • HTML • CSS
<code></code>	Ordered list, each item has a number.	<code></code> • HTML • CSS

Tables

Tag	Description	Example
<code><table></code>	Defines a table. By default a table has no borders and is only as wide as the text it contains.	
<code><tr></code>	Defines a table row (only has value within <code><table></code> tag). Table rows can contain either table data elements or table header cells.	
<code><td></code>	Table data element (standard table cell). Can contain many types of data including text, images, links, lists, or even another table.	<code><table></code> <code><tr></code> <code><td></code> cell 1 <code><td></code> cell 2 <code></tr></code> <code></table></code> cell cell 2
<code><th></code>	Table header cell (a table cell with bold text).	<code><table></code> <code><tr></code> <code><th></code> heading <code></tr></code> <code><tr></code> <code><td></code> content <code></td></code> <code></tr></code> <code></table></code> heading content

Input

Because the attributes used with input elements varies so much depending on the type of input you want to use, we have provided several specific examples of using different types of input.

Example	Description
<code><input type = "button" value = "change" onclick = "alert('clicked button')"></code>	<ul style="list-style-type: none">type is buttonvalue is text that appears on buttononclick is event handler, specifies to call alert function when button is clicked
<code><input type = "color" value = "#001A57" id = "clr" onchange = "docolor()"></code>	<ul style="list-style-type: none">type is color pickervalue is default color valueid lets us refer to input element in JavaScriptonchange is event handler, specifies to call docolor function when color is changed
<code><input type = "range" min = "10" max = "100" value = "10" id = "eldr" oninput = "dosquare()"></code>	<ul style="list-style-type: none">type is slidermin is minimum value, max is maximum valuevalue is default valueid lets us refer to input element in JavaScriptoninput is event handler, specifies to call dosquare function when slider is changed
<code><input type = "text" id = "finput"></code>	<ul style="list-style-type: none">type is textid lets us refer to input element in JavaScript
<code><input type = "file" multiple = "false" accept = "image/*" id = "finput" onchange = "upload()"></code>	<ul style="list-style-type: none">type is filemultiple = "false" indicates user cannot select multiple filesaccept = "image/*" indicates user can only select image filesvalue is default valueid lets us refer to input element in JavaScriptonchange is event handler, specifies to call upload function when input changes

CSS

Full list of CSS properties

Mozilla color picker tool

This website challenges people to use CSS to make as many different stylized versions as possible using the same HTML code.

Common CSS Properties and Values

Property	Example Values	Use with	Example
<code>color</code>	blue rgb(0,0,255) #0000FF	text: paragraphs, links, list elements, table cells, headings	<code>h1 { color: rgb(0,0,255); }</code>
<code>font-size</code>	12pt 16px 100%	text	<code>p { font-size: 14pt; }</code>
<code>text-align</code>	left right center justify	text	<code>td { text-align: center; }</code>
<code>background-color</code>	blue rgb(0,0,255) #0000FF	table, table cell, page backgrounds	<code>body { background-color: #00FF00; }</code>
<code>vertical-align</code>	top middle bottom	table cells	<code>th { vertical-align: top; }</code>
<code>float</code>	left right	images	<code>img { float: right; }</code>
<code>width</code>	100px	tables, table cells, images	<code>img { width: 80px; }</code>
<code>height</code>	100px	tables, table cells, images	<code>td { height: 10px; }</code>
<code>border-width</code>	5px	tables, table cells, images	<code>table { border-width: 2px; }</code>
<code>border-style</code>	solid dotted dashed	tables, table cells, images	<code>table { border-style: solid; }</code>
<code>border-color</code>	blue rgb(0,0,255) #0000FF	tables, table cells, images	<code>table { border-color: red; }</code>
<code>border</code>	5px 10px dotted 5px dashed green	tables, table cells, images	<code>table { border: 2px solid red; }</code>
<code>border-collapse</code>	collapse	table	<code>table { border-collapse: collapse; }</code>

Course Specific JavaScript Functions

SimplePixel

For these examples, assume

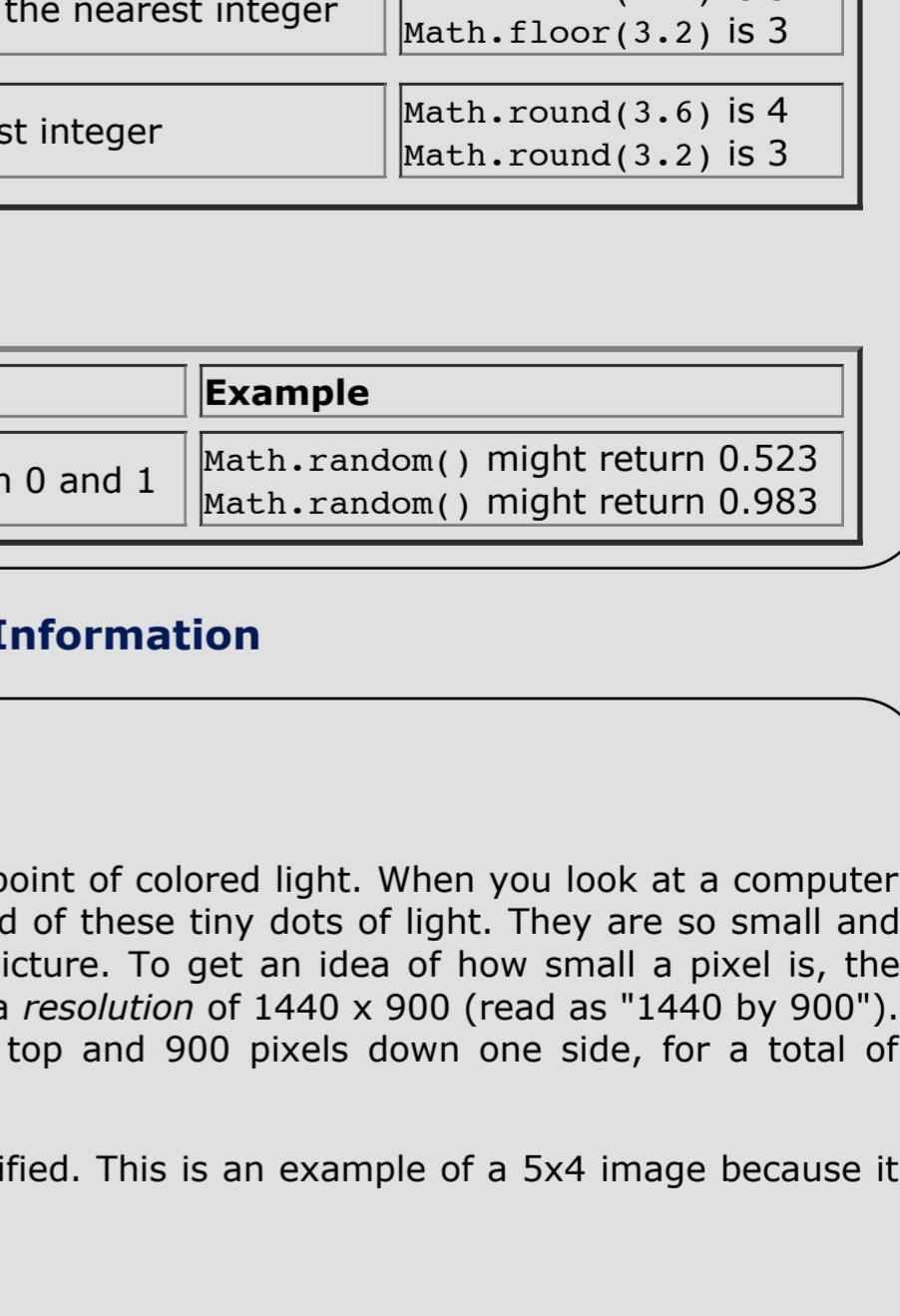
- pix1 is a pixel at coordinate (100, 200) representing the color Duke blue, with RGBA values of (0, 26, 87, 255)
- pix2 is a pixel at coordinate (300, 400) representing the color white, with RGBA values of (255, 255, 255, 255)

Function name	Description	Example
<code>getX()</code>	returns the pixel's x-coordinate within the image	<code>pix1.getX() is 100</code>
<code>getY()</code>	returns the pixel's y-coordinate within the image	<code>pix1.getY() is 200</code>
<code>getRed()</code>	returns the value of the pixel's red component (always in the range 0-255)	<code>pix1.getRed() is 0</code>
<code>getGreen()</code>	returns the value of the pixel's green component (always in the range 0-255)	<code>pix1.getGreen() is 26</code>
<code>getBlue()</code>	returns the value of the pixel's blue component (always in the range 0-255)	<code>pix1.getBlue() is 87</code>
<code>getAlpha()</code>	returns the value of the pixel's alpha, or transparency component (always in the range 0-255)	<code>pix1.getAlpha() is 255</code>
<code>setRed(newR)</code>	changes the value of the pixel's red component to newR if newR is not in the range of 0-255 it is changed to be in that range	<code>pix1.setRed(255) changes the color to (255, 26, 87, 255) red</code>
<code>setGreen(newG)</code>	changes the value of the pixel's green component to newG if newG is not in the range of 0-255 it is changed to be in that range	<code>pix1.setGreen(255) changes the color to (255, 255, 255, 255) white</code>
<code>setBlue(newB)</code>	changes the value of the pixel's blue component to newB if newB is not in the range of 0-255 it is changed to be in that range	<code>pix1.setBlue(255) changes the color to (255, 255, 255, 255) white</code>
<code>setAlpha(newA)</code>	changes the value of the pixel's alpha, or transparency component to newA if newA is not in the range of 0-255 it is changed to be in that range	<code>pix1.setAlpha(100) changes the color to (26, 100, 87, 100) light blue</code>
<code>setAllFrom(otherPixel)</code>	changes the value of all of the pixel's components (its red, green, blue, and alpha) to match otherPixel's values	<code>pix1.setAllFrom(pix2) changes the color of pix1 to (0, 26, 87, 255) blue</code>

SimpleImage

For these examples, assume the variable logo has the value of the image "devil.png" below. It is 100 pixels wide and 85 pixels tall.



Function name	Description	Example
<code>new SimpleImage(filename)</code>	creates a SimpleImage to represent the image in filename	<code>new SimpleImage("devil.png")</code>
<code>new SimpleImage(width, height)</code>	a SimpleImage whose dimensions are width by height. All the pixels in this image are black (0,0,0,255)	<code>new SimpleImage(100, 100)</code>
<code>new SimpleImage(fileInputElement)</code>	creates a SimpleImage to represent the image selected by the fileInputElement given from the web page	<code>var input document.getElementById("fileLoader"); var img = new SimpleImage(input);</code>  assuming the user selected that image from their computer.
<code>getWidth()</code>	returns the image's width, or number of pixels in the X direction	<code>logo.getWidth() is 100</code>
<code>getHeight()</code>	returns the image's height, or number of pixels in the Y direction	<code>logo.getHeight() is 85</code>
<code>getPixel(x,y)</code>	returns the pixel in this image at the coordinate (x, y)	<code>logo.getPixel(0, 0) is the pixel (255, 255, 255, 255)</code>
<code>setPixel(x,y,pixel)</code>	copies the RGBA values from the given pixel into pixel at the (x,y) coordinates given	<code>logo.setPixel(50, 42, pix2) changes the color to white</code>
<code>setSize(width, height)</code>	resizes the image to be width by height. The image is scaled to fit into the new dimensions.	<code>logo.setSize(300, 85) is</code> 
<code>values()</code>	returns all the pixels in the image, starting in the upper-left corner and moving down to the lower-right corner, providing a way to access each pixel in turn	<code>for (var pixel of logo.values()) { // modify pixel }</code>

Printing

Function name	Description	Example
<code>print(something)</code>	displays something in the main "See It" area of the page	<code>print(image)</code>